L Number	Hits	Search Text	DB	Time stamp
-	49639	synchroniz$4 with (time or (time adj4 day$1))	USPAT; US-PGPUB	2004/09/27 14:39
-	32156	(synchroniz$4 with (time or (time adj4 day$1))) and @ad<19981113	USPAT; US-PGPUB	2004/09/27 14:40
-	5778	((synchroniz$4 with (time or (time adj4 day$1))) and @ad<19981113) and timer$1	USPAT; US-PGPUB	2004/09/27 14:40
-	2480	synchroniz$4 with (time or (time adj4 day$1)) with (machines or computers or workstations)	USPAT; US-PGPUB	2004/09/27 14:40
-	1507	(synchroniz$4 with (time or (time adj4 day$1)) with (machines or computers or workstations)) and @ad<19981113	USPAT; US-PGPUB	2004/09/27 14:40
-	344	((synchroniz$4 with (time or (time adj4 day$1)) with (machines or computers or workstations)) and @ad<19981113) and timer$1	USPAT; US-PGPUB	2004/09/27 14:41
-	1	5990638.URPN.	USPAT	2004/09/27 15:39
-	6	("3624371" | "4611155" | "5218549" | "5990638" | "6037738" | "6188194").PN.	USPAT	2004/09/27 15:39

US-PAT-NO: 5990638

DOCUMENT-IDENTIFIER: US 5990638 A
See image for Certificate of Correction

TITLE: Synchronizing method for communication

---------- KWIC ---------

Application Filing Date - AD (1):
19980223

Brief Summary Text - BSTX (4):
In a system consisting of a numerical control device (hereinafter referred to as CNC device), a robot-controller and its peripheral devices or similar industrial device, it is sometimes necessary to link a plurality of devices by means of serial communication for the purpose of ensuring that they operate in synchronization with one another. For instance, **machine**-tools which are driven and controlled by CNC devices must be **synchronized** in order to detect the current position of the individual feed shafts at the same point in **time**. Similarly, the individual shafts have to be moved in synchronization. Signals for the purpose of implementing synchronization of this sort are transmitted by way of serial communication.

Brief Summary Text - BSTX (5):
For instance, FIG. 13 illustrates a system where two machines (master and slave) are connected by communication. One method of controlling the slave **machine by synchronizing** it with the aid of serial communication to the master **machine** involves emitting from the master **machine** to the slave **machine** a signal requesting transmission of a **synchronizing** signal with a predetermined cycle from a **timer at a time** which is to form the basis for synchronization, transmitting a packet (frame) on a transmission control circuit identifying it as a **synchronizing** signal, and transmitting this packet after conversion to a serial signal by means of a parallel-serial converter. Each time the reception control circuit of the slave machine receives this packet by way of a serial-parallel converter, it either operates its own internal sequencer or controls its own internal **timer** in such a manner as to make it conform to the **timer** of the master machine. This method is well known.

Brief Summary Text - BSTX (8):
FIG. 15 is a diagram which elucidates this discrepancy in timing for the example shown in FIG. 13. In FIG. 15 only the header section is treated as synchronization data (synchronizing signal). It is assumed that a request for transmission is generated cyclically in accordance with an internal **timer** in the master machine, and that now at a given time (reference time t0) such a request for transmission has been generated from the **timer** of the master

machine. The transmission control circuit transmits a packet of the type illustrated in FIG. 14 to the parallel-serial converter, where it is converted into a serial signal and transmitted to the slave machine. There is a delay D1 from the time at which this signal requesting transmission was generated until the packet is transmitted to the transmission line. The reception of this packet by the slave machine entails further delays in the forms of the propagation time on the transmission line Dp and the time D2 required for synchronization. Inasmuch as the slave **machine** becomes aware of the arrival of the header section only after a fixed interval D3 from the completion of its reception, the **synchronizing** signal (header detection signal) generated by the slave **machine** is subject to a delay equivalent to the total of the delays (D1+Dp+D2+D3) from the **time** t0 at which the request for transmission was generated by the master **machine**.

Brief Summary Text - BSTX (9):

Conventionally, the propagation delay time Dp during which the signal propagates along the transmission line has been regarded as capable of being ignored. The remaining delay times D1, D2 and D3 are always constant and can be determined without difficulty from the circuitry configuration which was adopted at the time of designing the master and slave machines. Thus, when synchronizing in the conventional manner by the internal **timer** in the slave machine, the delay (D1+D2+D3) determined from the circuitry configuration is previously set in the slave machine as a correction value. When the slave machine detects the synchronizing signal (header detection signal), it sets the correction value as the initial value in the internal **timer** to activate the **timer**. The **timer** takes `time up` (the time when the internal **timer** in the master machine has measured the value obtained by subtracting the correction value from the time beginning with the reference time t0 and ending with synchronization time T) as the synchronization time, thereby accomplishing desired synchronization by making correction corresponding to the delay. The opposite method is also used, whereby the delay is corrected by advancing the timing with which transmission begins from the master **machine** by the known amount of the delay and controlling the internal **timer** in the slave **machine**, by regarding the **time at which the synchronizing** signal (header detection signal) is detected as reference **time**.

Brief Summary Text - BSTX (14):

With a view to achieving this object, the present invention is a method of **synchronizing** a plurality of **machines** connected by serial communication and operating in synchronization, wherein the propagation delay **time** required for the synchronization signal to be propagated from the transmitting side to the receiving side is programmed as a correction value into the **machine** on either the transmitting or receiving side, so that the plurality of **machines** can be **synchronized** by correcting the **synchronizing** signal on the basis of the correction value.

Detailed Description Text - DETX (7):

The CNC device 10 has a processor 101, while a ROM 102 in which the system program is stored, a RAM (comprising in part a non-volatile RAM) 103 in which the NC program and various other data are stored and which is used for

processing and other purposes, a transmission control circuit 104 and a reception control circuit 105 are connected by buses. A **timer** circuit 108 outputs with a predetermined cycle to the processor 101 a signal S1 for calculation of the command voltage to begin, and after allowing ample time for the processor 101 to finish calculating the command voltage, outputs with the same predetermined cycle to the transmission control circuit 104 a signal S2 for starting transmission of the command voltage to the servoamplifiers SA1, SA2. There is also connected to the transmission control circuit 104 a parallel-serial converter 106 whereby parallel signals are converted into serial signals, transmission data which has been converted in this manner being sent to the transmission line L1. Reception data input by way of the transmission line L1 is input into a serial-parallel converter 107, where serial signals are converted into parallel, and this serial-parallel converter 107 is connected to the reception control circuit 105.

Detailed Description Text - DETX (8):

The principal section of the servoamplifier SA1 will be explained using the block diagram in FIG. 3. Data sent from the CNC device 10 by way of the transmission line L1 is input into a serial-parallel converter 201 and into a buffer circuit 203. This buffer circuit 203 is connected to the transmission line L2, and data received from the CNC device 10 is transmitted on to the next servoamplifier SA2. A reception control circuit 202 is connected to the serial-parallel converter 201, and data which has been converted into a parallel signal is input into this reception control circuit 202. To this reception control circuit 202 is also connected a digital-analogue converter 204, data output by the reception control circuit 202 being converted into an analogue signal in this digital-analogue converter 204 to become a command voltage to the servo motor M1. The reception control circuit 202 is connected to a **timer** circuit 206, to which a header detection signal (synchronizing signal) S3 is output in order to start this **timer** circuit 206. As is explained below, a value obtained by adding pre-determined correction value 1 and correction value 2 in an adder 205 is programmed into the **timer** circuit 206 as the correction value of the synchronizing signal.

Detailed Description Text - DETX (9):

The **timer** circuit 206 outputs a latch and transmission timing signal S4 to a data latcher 210 and a transmission control circuit 211, and the value of reversible counter 207 which counts feedback pulses from the pulse coder P1 is latched in the data latcher 210. The transmission control circuit 211 outputs this latched value and transmission data from the downstream servoamplifier SA2, which is input by way of a serial-parallel converter 208 and stored in a buffer circuit 209, to a parallel-serial converter 212, where it is converted it into a parallel signal and sent to the transmission line L1, being output to the CNC device as feedback data (data indicating the positions of the servo motors M1, M2).

Detailed Description Text - DETX (12):

The **timer** circuit 108 within the CNC device 10 outputs to the processor 101 with a predetermined cycle a signal S1 for starting calculation of the command voltage. The timing with which this signal Si for calculation of the command voltage to begin is generated is taken as the reference time t0. When the

processor 101 receives the signal S1 for starting calculation of the command voltage, it reads and analyses the NC program within the RAM 103, and on the bases of movement commands already acquired and feedback data from the servoamplifiers SA1, SA2 which has been input into the reception control circuit 105 by way of the serial-parallel converter 107, processes the feedback on position and speed, calculates a new command voltage for servo motor of each shaft within the predetermined cycle, and outputs to the transmission control circuit 104. The signal S2 for transmission of the command voltage to the servoamplifiers to begin is generated, as shown in FIG. 4, from the **timer** circuit 108 at a predetermined delay D0 in relation to the reference time t0, and input into the transmission control circuit 104 is effected. When the transmission control circuit 104 receives the signal S2 for starting transmission of the command voltage, as FIG. 5 shows, it sends to the parallel-serial converter 106 for conversion into a serial data a packet, wherein a header which functions as a synchronizing signal is added to the data for the servoamplifiers SA1, SA2, and transmits the serial data by transmission line 1.

Detailed Description Text - DETX (13):

When the servoamplifier SA1 receives this packet, it transmits the packet on to the downstream servoamplifier SA2 by way of the buffer circuit 203, and also outputs the packet to the reception control circuit 202, after converting it into a parallel signal in the serial-parallel converter 201. When the reception control circuit 202 reads the header of the packet, as is shown in FIG. 4, it outputs the header detection signal (synchronizing signal) S3 to the **timer** circuit 206, and also outputs reception data for its own servoamplifier SA1 to the digital-analogue converter 204, where it is converted into analogue voltage and output to the servo motor M1. The address of the data in the packet is determined by its storage position (bit), and in this embodiment the data for the second servoamplifier SA2 is stored in the section of a set number of words following the header section, the data for the first servoamplifier is stored in the section of a set number of words following that. The servoamplifiers SA1, SA2 respectively read the corresponding locations in the packets as data for themselves.

Detailed Description Text - DETX (14):

As has already been pointed out, the delay **time** D1 from the generation of the signal S2 for starting transmission of the command voltage to the servoamplifiers until the data is serialised and the packet transmitted to the transmission line, the delay **time** D2 for **synchronizing** serial data on the reception side, and the delay **time** D3 until the completion of header reception is detected are already known at the stage of designing the circuitry configurations of the **machines** which make up this system. This means that their total (D1+D2+D3) is also known, as is the delay time D0 from the generation of the signal S1 for starting calculation of the command voltage (reference time t0) to that of the signal S2 for starting transmission of the command voltage to the servoamplifiers. The timing for reading position data on the servo motors SA1, SA2 has been determined in advance as a point when time T has elapsed from the reference time t0, so that the value T is also known. Thus, the above known delay time (D0+D1+D2+D3) against the elapsed time T of the timing for reading this known position data is programmed as correction value 1. The delay time Dp which is required for the signal to

propagate on the transmission line is determined by the system, and becomes known only when the type (material) and length of the transmission line is determined. This propagation delay time Dp is therefore programmed as correction value 2 once the system is determined.

Detailed Description Text - DETX (15):

Correction value 1 and correction value 2 are added in the adder 205, and the resultant value [D0+D1+D2+D3+Dp] is determined. This value is programmed as the initial value of the **timer** circuit 206. The **timer** circuit 206 begins measuring once the header detection signal (synchronizing signal) S3 is received from the reception control circuit 202, and, when the time T is reached, outputs the latch and transmission timing signal S4 to the data latcher 210 and transmission control circuit 211. In other words, the time from reception of the header detection signal S3 to output of the latch and transmission timing signal S4 is T-[D0+D1+D2+D3+Dp].

Detailed Description Text - DETX (17):

At the point in time when the reception control circuit 202 reads the header section and transmits the header detection signal (synchronizing signal) S3 to the **timer** circuit 206, there is already a delay of [D0+D1+D2+D3+Dp] in relation to the reference time t0. Since the **timer** circuit 206 measures the set value T-[D0+D1+D2 +D3+Dp] before outputting the latch and transmission timing signal S4 and latching the position of the servo motor, the time of this latching is delayed by a total of the abovementioned delay time and the time measured by the **timer** circuit. In other words,

Detailed Description Text - DETX (28):

This differs from the first embodiment only in that it has a second **timer** circuit 109, which is connected to the reception control circuit 105, while a delay time measurement packet transmission signal S5 is input from the processor 101. Those parts of the configuration which are the same as those shown in FIG. 2 are denoted with the same reference numerals.

Detailed Description Text - DETX (30):

The difference in structure between the servoamplifier SA1 illustrated in FIG. 8 and that illustrated in FIG. 3 is that the former has an additional second **timer** circuit 213 and a delay time measurement packet detection circuit 214. The input side of the delay time measurement packet detection circuit 214 is connected to the serial-parallel converter 208 which receives data from the downstream servoamplifier SA2, while its output side is connected to the second **timer** circuit 213, to which it outputs a delay time measurement packet detection signal S7 from the downstream servoamplifier. The second **timer** circuit 213 is connected in such a manner as to receive a delay time measurement packet reception signal S6 from the reception control circuit 202, while this delay time measurement packet reception signal S6 is also input into the transmission control circuit 211. The output of the second **timer** circuit 213 is fed to the transmission control circuit 211. The above are the differences. That parts of the configuration which are the same as those shown in FIG. 3 are denoted by the same reference numerals.

Detailed Description Text - DETX (32):

Before normal communication commences, a delay time measurement command is input, and the processor 101 of the CNC device 10 outputs the delay time measurement packet transmission signal S5 to the transmission control circuit 104 and the second **timer** circuit 109. When the second **timer** circuit 109 receives this signal, it begins measuring. Meanwhile, the transmission control circuit 104, on receiving this signal, outputs the delay time measurement packet to the transmission line L1 by way of the parallel-serial converter 106. The delay time measurement packet and the normal data packet are distinguished by the header data.

Detailed Description Text - DETX (33):

The upstream servoamplifier SA1, having received the delay time measurement packet, transmits it on by way of the buffer circuit 203 to the downstream servoamplifier. This process is repeated down to the servoamplifier furthest downstream. Each servoamplifier receives the header section of the delay time measurement packet by way of the serial-parallel converter 201 into the reception control circuit 202. The reception control circuit 202 outputs the delay time measurement packet reception signal S6 to the second **timer** circuit 213 and transmission control circuit 211. Upon receiving this signal S6, the second **timer** circuit 213 begins measuring, while the transmission control circuit 211 sends a reply delay time measurement packet by way of the parallel-serial converter 212 to the transmission line and thus to the CNC device 10 or the upstream servoamplifier.

Detailed Description Text - DETX (34):

When the reception control circuit 105 detects the header of the delay time measurement packet by way of the serial-parallel converter 107, the CNC device 10 outputs a signal to the second **timer** circuit 109 in order to halt its measurement. The count value of the second **timer** circuit 109 is the total of the known delay times resulting from the serialisation and synchronization of the above data [D1+D2+D3] and the propagation time Dp generated in both directions on the transmission line (L1 in FIG. 1) between the CNC device 10 and the servoamplifier furthest upstream. The propagation delay time Dp can be determined by calculation from the count value of the second **timer** circuit 109 and the known delay time [D1+D2+D3]. The propagation delay time Dp determined in this manner is sent to the servoamplifier SA1 furthest upstream from the CNC device and programmed in as correction value 2.

Detailed Description Text - DETX (35):

The delay time measurement packet detection circuit 214 of each servoamplifier, having received from the servoamplifier next downstream from itself the header section of the reply delay time measurement packet by way of the serial-parallel converter 208, transmits the delay time measurement packet reception signal S7 to the second **timer** circuit 213 in order to halt its measurement. The reply delay time measurement packet received from the downstream servoamplifier is not passed on to the upstream servoamplifier, but discarded. As a result, the value which is stored in the second **timer** circuit 213 of each servoamplifier represents the time from the generation of the delay time measurement packet reception signal S6 from the reception control circuit

202 in the servoamplifier in question to the generation of the delay time measurement packet reception signal S7 from the delay time measurement packet detection circuit 214 after reception of the header section of the reply delay time measurement packet from the servoamplifier next downstream.

Detailed Description Text - DETX (36):

Next, the CNC device 10 transmits a packet to each servoamplifier commanding it to transmit the measurement value in its second **timer** circuit 213. Upon receiving this packet, the reception control circuits 202 of the servoamplifiers send a transmission command S8 to the transmission control circuit 211, and the value in the second **timer** circuit 213 is transmitted to the CNC device 10.

Detailed Description Text - DETX (37):

Let it be assumed here that a plurality of servoamplifiers SA1, SA2 . . . SAn is connected to the CNC device 10 in the manner of a daisy-chain. If so, the measurement values in the second **timer** circuits 213 of the servoamplifiers will be as follows.

Detailed Description Text - DETX (38):

Let t1 be the time at which the delay time measurement packet from the device next upstream from the first servoamplifier SAi (servoamplifier i-1, or the CNC device 10 if i=1) has been received by amplifier SA; (this being the time at which it is input into the serial-parallel converter 201 and buffer circuit 203). Let t2 be the time from t1 to the halting of the measurement by the second time circuit 213, the delay time measurement packet having been transmitted from the servoamplifier SAi to the servoamplifier SAi+1 next downstream, the servoamplifier SAi+1 next downstream having received this signal and transmitted the reply delay time measurement packet, and the servoamplifier SAi having received it. Furthermore, let t3 be the time from t1 to the commencement of measurement by the second **timer** circuit 213, the servoamplifier SAi having detected reception of the delay time measurement packet from the machine next upstream. In this case, the measurement value of the second **timer** circuit 213 is [t2-t3]. In other words, t2 is the sum of the processing time in the buffer circuit 203 of the servoamplifier SAi, the propagation delay time on the transmission line from the servoamplifier SAi to the servoamplifier SAi+1 next downstream, the delay time from the header section of the delay time measurement packet being input into the serial-parallel converter 201 in the servoamplifier SAi+1 next downstream to the delay time measurement packet reception signal S6 being output from the reception control circuit 202 (i.e., processing time of the S/P converter 201 and reception control circuit 202), the processing time in the transmission control circuit 211 and parallel-serial converter 212, the propagation delay time between the servoamplifier SAi+1 next downstream and the servoamplifier SAi, the processing time in the serial-parallel converter 208 of the servoamplifier SAi, and the processing time in the delay time measurement packet detection circuit. Moreover, t3 is the total of the processing time in the serial-parallel converter 201 and reception control circuit 202 of the servoamplifier SAi.

Detailed Description Text - DETX (39):

Of these, all the values with the exception of the propagation time on the transmission line are known, having been clarified at the time when the circuitry configuration was designed. It is therefore possible to determine the propagation delay time on the transmission line between the servoamplifiers from the measured values of each of the second **timer** circuits and these known values. For the servoamplifier SA2 next downstream from the servoamplifier SA1 furthest upstream, correction value 2 is set, by the CNC device 10, as being the sum of the propagation delay time resulting from the transmission line between the CNC device and the servoamplifier SA1 furthest upstream (the value previously determined as correction value 2 for the most upstream servoamplifier) and the propagation delay time resulting from the transmission line between the servoamplifier SA1 furthest upstream and the servoamplifier SA2 next downstream. For the servoamplifier SA3 next-but-one downstream from the servoamplifier SA1 furthest upstream, a value obtained by adding to the value, being determined as the correction value 2 for the servoamplifier SA2 next downstream from the servoamplifier SA1 furthest upstream, the propagation delay time, resulting from the transmission line between the servoamplifier SA2 next downstream and the servoamplifier SA3 next-but-one downstream, may be programmed as correction value 2. This correction value 2 is programmed in the same way against successive servoamplifiers.

Detailed Description Text - DETX (40):

In this manner it is possible for the propagation delay time Dp to be measured and corrected automatically even where a plurality of servoamplifiers is connected in the manner of a daisy-chain. If the servoamplifiers are connected not in the manner of a daisy-chain but each by means of its respective transmission line to the CNC device, there is no need for a buffer circuit 203, serial-parallel converter 208, buffer circuit 209, delay time measurement packet detection circuit 214 or second **timer** circuit 213 in each servoamplifier. Instead, the delay time measurement packet can be transmitted to each servoamplifier, the length of time until detection of the header of the reply delay time measurement packet is measured, and the propagation delay time between the CNC device and each servoamplifier is determined. In other words, it may be determined by the method whereby the propagation delay time to the servoamplifier furthest upstream was determined above.

Detailed Description Text - DETX (44):

Furthermore, as is shown in FIG. 10 and will be clear from a comparison with FIG. 2 which illustrates the CNC device to which the first embodiment is applied, the CNC device to which the third embodiment is applied is characterised by the fact that the signal S1 for starting calculation of the command voltage is output from the **timer** circuit 108 as the synchronizing signal SS. Moreover, as will be seen from FIG. 3, in the servoamplifier SA1, to which the first embodiment is applied, the header detection signal (synchronizing signal) S3 from the reception control circuit 202 is input into the **timer** circuit 206. However, as FIG. 11 shows, in the servoamplifier SA1, to which the third embodiment is applied, the synchronizing signal SS is input from the CNC device. That part of the configuration which is the same in FIGS. 9-11 as that shown in FIGS. 1-3 has been allocated the same reference numerals.

Detailed Description Text - DETX (46):

The signal S1 for calculation of the command voltage to begin is output from the **timer** circuit 108 of the CNC device 10 with a predetermined cycle. The processor 101 receives this signal and begins calculating the command voltage. The signal S2 for starting transmission of the command voltage is generated with the same predetermined cycle from the **timer** circuit 108 but delayed in relation to the signal S1 for starting calculation of the command voltage. This causes the command voltage packet to be transmitted by way of the transmission control circuit 104 and parallel-serial converter 106 to the servoamplifiers SA1, SA2, in the same manner as is shown with respect to the first embodiment in FIGS. 1-3.

Detailed Description Text - DETX (48):

The servoamplifiers SA1, SA2 receive this synchronizing signal SS, causing the **timer** circuit 206 to start up. Into this **timer** circuit 206 has been programmed in advance a correction value obtained by adding correction value 1 and correction value 2 in the adder 205. When each **timer** circuit 206 reaches the fixed time T, the latch and transmission signal S4 is output to the data latcher 210 and transmission control circuit 211. This signal allows the positions of the servo motors M1, M2 to be transmitted from the parallel-serial converter 212 to the CNC device 10. As in the first embodiment illustrated in FIGS. 1-3, in this third embodiment too, it is necessary to detect the positions of the servo motors M1, M2 at a point in time after the passage of the time T from when the signal S1 for starting calculation of the command voltage was generated (t0).

Detailed Description Text - DETX (49):

A delay of several clock pulses is generated in the synchronizing signal SS when it is converted within the servoamplifiers into a signal synchronized with the internal clock. This delay is already a known factor at the design stage, and the corresponding correction value is therefore programmed as correction value 1, while the propagation delay time between the CNC device and each of the servoamplifiers is programmed as correction value 2. As a result, (correction value 1+correction value 2) is programmed into each **timer** circuit 206 as the initial value.

Detailed Description Text - DETX (52):

In each of the above embodiments, correction values 1 and 2 have been added in the adder 205 and programmed as the correction value into the **timer** circuit 206, but when determining correction value 2 it is also possible to do so in consideration of correction value 1 and program this value into each of the servoamplifiers. In this case there is no need for the adder 205. If the configuration is not in the manner of daisy-wheel, and the CNC device and the servoamplifiers are connected by separate transmission lines, the above correction value can be set on the CNC side and taken into consideration when transmitting the synchronizing signal from the CNC device 10 to each of the servoamplifiers.

Claims Text - CLTX (2):

programing the propagation delay **time** required for a **synchronizing** signal to

propagate from the transmission side to the reception side into the **machine** on the transmission or reception side as a correction value; and

Claims Text - CLTX (8):
6. The method of synchronization in communication according to claim 1, wherein the delay **time in the machine** elements on the transmission line which is required for the **synchronizing** signal to propagate is programmed beforehand into the **machine** on said transmission or reception side, and at the stage when said plurality of **machines** is connected by the transmission line, the delay **time** resulting from the transmission line is programmed into the **machine** on said transmission or reception side, the correction value being determined from these two delay **times** to correct both the **synchronizing** signal, and the propagation delay **time, thus synchronizing** said plurality of **machines**.

Claims Text - CLTX (14):
a first **timer** which begins measuring when said means of transmission transmits said specific signal, and finishes measuring when said means of reception receives from the second machine the answer signal to the effect that it has received said specific signal,

Claims Text - CLTX (17):
a second **timer** for the purpose of adjusting the time from the request for transmission from the first machine until sampling of the signal to the first machine in response to the request,

Claims Text - CLTX (19):
means for calculating propagation delay time, whereby the propagation delay time between said first machine and said second machine is calculated from the measured time of said first **timer**; and

Claims Text - CLTX (20):
means for programming the value calculated by said calculating propagation delay time calculating means as one correction value of the second **timer** of the second machine.

Claims Text - CLTX (24):
a **timer** which begins measuring when said reception control device receives a signal from said numerical control device, and

Claims Text - CLTX (25):
a transmission control circuit which sends feedback data from pulse coders attached to the servo motor and servo motors downstream from it when the **timer** has finished measuring,

Claims Text - CLTX (26):

further said **timers**

Claims Text - CLTX (27):
connect a means whereby a value obtained by subtracting the time required for the signal to propagate on the transmission line from the time required from the output of the signal for starting calculation of the command voltage within said numerical control device until the reception of the signal by the **timer** from said reception control circuit is set as a first correction value, and a means whereby the time required for the signal to propagate on the transmission line is set as a second correction value, and,

Claims Text - CLTX (33):
a first **timer** which begins measurement with the transmission of said delay time measurement packet, and finishes measurement when said reception control circuit receives the delay time measurement packet sent back from the servoamplifier furthest upstream,

Claims Text - CLTX (34):
a first delay time calculating means whereby the propagation delay time on the transmission line between this numerical control device and the servoamplifier furthest upstream is calculated on the basis of the value measured by said first **timer**; and

Claims Text - CLTX (39):
a second **timer** which begins measurement when said reception control circuit receives the header of the delay time measurement packet, and finishes measurement when said delay time measurement packet detection circuit detects the header of the delay time measurement packet which has been sent back,

Claims Text - CLTX (40):
a second delay time calculating means whereby the propagation delay time on the transmission line between this servoamplifier and the servoamplifier next downstream is calculated on the basis of the value measured by said second **timer,** and,

US-PAT-NO: 6725278

DOCUMENT-IDENTIFIER: US 6725278 B1

TITLE: Smart synchronization of computer system time clock based on network connection modes

---------- KWIC ---------

Abstract Text - ABTX (1):

A method and apparatus are provided for performing synchronization of a time clock maintained by a computer system based on the network connection modes of the computer system. The technique allows the **time** clock of the **computer** system to be **synchronized** at the earliest opportunity that will not result in a disruption or inconvenience to the user. When synchronization of the time clock is required, and the computer system does not already have an active network connection, the computer system queries a configuration database to determine its default network connection mode. If the default connection mode is potentially user disruptive (e.g., use of a dial out modem is required), synchronization of the time clock is not performed until an active network connection has been established. If the default connection mode is not likely to be user-disruptive (e.g., doesn't require a modem connection), then a network connection is immediately established to synchronize the time clock.

Application Filing Date - AD (1):

19980917

Brief Summary Text - BSTX (4):

Most conventional computer systems maintain an internal clock to keep track of the time of day. Accurate time of day measurements are required in a wide variety of applications, such as managing and tracking electronic mail (email), timing back-ups of data on a network, synchronizing communications between clients and servers, and managing multimedia teleconferences. Because the **time** clocks maintained by most **computer** systems tend to be subject to a certain amount of error, or "drift", it becomes necessary to **synchronize** such clocks to an accurate reference clock from **time to time,** to maintain their accuracy.

Brief Summary Text - BSTX (5):

Various solutions have been developed to **synchronize the time** clock of a **computer** system. A simple method is for the user of a computer system to manually adjust the clock whenever the clock appears to have drifted. This technique, however, is both inconvenient for the user and subject to its own inaccuracies. A more sophisticated solution makes use of a server computer system operating on a network, such as the Internet. The server maintains a highly accurate time clock, such as an atomic clock, and provides accurate time readings to other computer systems on the network using a communication protocol such as Network Time Protocol (NTP). A client **computer** system may

send a request for an accurate **time** reading via the Internet to an NTP server when it is necessary to **synchronize** its internal clock. The request may be routed to one of a number of secondary servers that function as intermediaries between client systems requiring clock synchronization and a primary NTP server. The use of such secondary servers is intended to reduce the loading on the primary NTP server. A primary NTP server may be maintained, for example, by a government entity, such as the U.S. Navy, while access to the primary NTP server is regulated by secondary NTP servers maintained by universities and business enterprises, for use by their students and employees, respectively.

Drawing Description Text - DRTX (7):

FIG. 4 is a flow diagram illustrating a routine for **synchronizing a time** clock based on the network connection modes of the **computer** system.

Detailed Description Text - DETX (2):

A method and apparatus for performing synchronization of a computer system time clock based on the computer system's network connection modes are described. The described method and apparatus allow the **time** clock of a **computer** system to be **synchronized** at the earliest opportunity that will not result in a disruption or inconvenience to the user, while avoiding unnecessary delays in **synchronizing** the clock. When synchronization of the time clock is required, and the computer system does not already have an active network connection, the computer system queries a configuration database to determine its default network connection mode. If the default connection mode is potentially user disruptive, synchronization of the time clock is not performed until an active network connection has been established. If the default connection mode is not likely to be user disruptive (i.e., is transparent to the user), then a network connection is immediately established to synchronize the time clock.

Detailed Description Text - DETX (12):

Network application software 23 is, for purposes of this description, is assumed to be an NTP compliant software application. Accordingly, network application software 23 is configured to, when executed by the client **computer** system 1, cause the client system 1 to connect to **time** server 2 via its network connection 6, to obtain an accurate **time** reading for purposes of **synchronizing** the internal **time** clock of client **computer** system 1.

Detailed Description Text - DETX (13):

According to one embodiment, the client **computer** system 1 further maintains at least two additional software components associated with synchronization of its **time** clock, namely, clock **synchronizer** 26 and network setup component 28. In general, clock **synchronizer** 26 configures the client **computer** system 1 to determine how often its internal **time** clock requires synchronization and to determine, when synchronization is required, whether or not it is appropriate to access the network immediately to perform synchronization. The latter determination is made based on information stored in an internal configuration database 30 maintained by client computer system 1. Stored in the configuration database 30 are data indicating the network connection modes that

are available to this client computer system 1, as discussed in greater detail below. Clock synchronizer 26 may be implemented as a software application, which operates in the background to other applications. Alternatively, clock synchronizer 26 may be implemented in any of various other forms, such as a component of the network application software 23 or the operating system 26, in hardware, or a combination thereof.

Detailed Description Text - DETX (19):

FIG. 4 illustrates a process that may be implemented by clock synchronizer 26, to perform clock synchronization in accordance with the foregoing description. At 401, when it has been determined that it is **time to synchronize the time** clock, then it is next determined at 402 whether the client **computer** system 1 has an active network connection. If the client computer system currently has an active network connection, then the routine proceeds to 406. At 406, the time clock is synchronized by accessing the time server 2 to obtain an accurate time reading. Accessing of the time server 2 may be initiated by clock synchronizer 26 directly, or by clock synchronizer's signaling the network application software 23 to do so.

Detailed Description Text - DETX (22):

In accordance with the present need-based synchronization technique, clock synchronizer 26 maintains a history of the amount of drift in the time clock of the client system 1 and determines, based on the history, how often the time clock requires synchronization. The time clock is then synchronized only as often as necessary, as determined by this computation. Thus, a **computer** system with a small amount of drift in its **time** clock will access the **time** server to **synchronize its time** clock less frequently then a **computer** system with a large amount of drift in its **time** clock.

Detailed Description Text - DETX (23):

FIG. 5 illustrates a routine, which may be embodied in clock synchronizer 26, for performing need based clock synchronization as described above. Initially, a minimum acceptable clock accuracy (threshold) is defined at 501. The threshold may determined based on the particular requirements of the machine and its intended uses. Next, at 502 the **time** server 2 is accessed to obtain a **time** reading and the internal **time** clock of the **computer** system is **synchronized** based on the **time** reading. This operation (502) may be performed subject to the constraints described above regarding avoiding disruption to the user. If 502 represented the first instance that clock synchronization was performed by this system, then the routine proceeds to 509, at which the time read back from the server is stored. After storing the **time,** the client **computer** system 1 waits for a predetermined period of **time,** which may be arbitrary, before **synchronizing** the clock again at 502. After performing a synchronization at 503 other than the first synchronization, the amount of drift .DELTA.T in the clock since the previous synchronization is then computed at 504. At 505, optionally, a history of multiple .DELTA.T values for the client computer system 1 is updated. Updating the history may include updating a computation of the average of all previous .DELTA.T values or some other similarly indicative value. At 506, the next synchronization time is determined based on the defined threshold and based on the computed .DELTA.T,

the history of .DELTA.T data (if any), or both. In this context, determining the next synchronization "time" may be interpreted to mean computing the amount of time until the next synchronization (i.e., the synchronization interval) or computing the specific time and/or date of the next synchronization. Smaller .DELTA.T values will result in greater amounts of time being allowed until the next synchronization, and vice versa. After determining the next synchronization time, then when it is determined to be time to resynchronize the clock based on the computation of 506, the routine is repeated from 502.

Detailed Description Text - DETX (28):

Clock **synchronizer** 26 may be configured to make use of other timing services that are available to the client **computer** system 1, which are as precise as the **time** server's clock. Such services are commonly included as part of the operating system or other software of many conventional computer systems and may be used for various precise timing functions. Assume, for example, that the time clock of a client computer system keeps time in seconds, but the client computer system maintains a local timing service which keeps time in microseconds and is independent of the time clock. A reading of N seconds from the time clock of the client computer system does not necessarily equate to N.times.10.sup.6 microseconds from the timing service. Accordingly, referring to FIGS. 7A and 7B, the following routine may be performed to synchronize the time clock more precisely.

Claims Text - CLTX (8):

8. A computer-implemented method of controlling synchronization of a time clock maintained by a computer system, the method being associated with a clock synchronization process of the computer system, the method comprising: determining whether a default network connection mode of the computer system requires a user-disruptive action; if the default network connection mode requires a user-disruptive action, then when the **computer** system does not have an active connection to a network used to **synchronize the time** clock, waiting until a process other than the clock synchronization process has established an active connection to the network for the **computer** system, before accessing a server on the network to **synchronize the time** clock; and if the default connection mode does not require a potentially user-disruptive action, then: immediately establishing an active connection to the network; and accessing the server to synchronize the time clock.

Claims Text - CLTX (14):

14. A **computer**-implemented method of **synchronizing a time** clock maintained by a **computer** system, the method comprising: when a synchronization of the **time** clock is to be performed, determining whether the **computer** system has an active connection to a network; if the **computer** system has an active connection to the network, then accessing a server on the network to **synchronize the time** clock; if the computer system does not have an active connection to the network, then: determining whether a default network connection mode of the computer system is user-disruptive; and if the default network connection mode is user-disruptive, then: waiting until the computer system has an active connection to the network; and accessing the server to **synchronize the time** clock only when the **computer** system has an active connection to the network;

and if the default connection mode is not user-disruptive, then: immediately establishing an active connection to the network; and accessing the server to synchronize the time clock.

Claims Text - CLTX (15):

15. A **computer**-implemented method of **synchronizing a time** clock maintained by a **computer** system, the method comprising: determining whether the **computer** system has an active connection to a network each **time** a synchronization of the **time** clock is to be performed; if the **computer** system has an active connection to the network, then accessing a server on the network to **synchronize the time** clock; if the computer system does not have an active connection to the network, then: determining whether a default network connection mode of the computer system requires activation of a modem by querying a configuration database of the processing system; if the default network connection mode requires activation of a modem, then waiting until the **computer** system has an active connection to the network to access the server to **synchronize the time** clock; and if the default connection mode does not require activation of a modem, then: immediately establishing an active connection to the network; and accessing the server to synchronize the time clock.

Claims Text - CLTX (16):

16. A **computer**-implemented method of **synchronizing a time** clock maintained by a **computer** system, the method comprising: when a synchronization of the **time** clock is to be performed, determining whether the **computer** system has an active connection to a network; if the **computer** system has an active connection to the network, then accessing a server on the network to **synchronize the time** clock; if the computer system does not have an active connection to the network, then: determining whether a default network connection mode of the computer system is user-disruptive; and if the default network connection mode is user-disruptive, then: waiting until the computer system has an active connection to the network; and accessing the server to **synchronize the time** clock only when the **computer** system has an active connection to the network; and if the default connection mode is not user-disruptive, then: immediately establishing an active connection to the network; and accessing the server to synchronize the time clock.

Claims Text - CLTX (17):

17. A computer system comprising: a processor; a communication device coupled to the processor for enabling the computer system to communicate on a network; and a storage device coupled to the processor, the storage device having instructions stored therein which configure the processor to: determine whether a default network connection mode of the computer system is user-disruptive; if the default network connection mode is user-disruptive, then when the **computer** system does not have an active connection to the network, wait until the **computer** system has an active connection to the network to access the network to **synchronize a time** clock maintained by the **computer** system; and if the default connection mode is not user-disruptive, then: immediately establish an active connection to the network; and access the network to synchronize the time clock.

Claims Text - CLTX (29):

29. A data signal embodied in a propagation medium, the data signal representing a plurality of instructions which, when executed on a computer system, cause the computer system to: determine whether the computer system has an active connection to a network when a synchronization of a time clock maintained by the computer system is to be performed; if the **computer** system has an active connection to the network, then access the network to **synchronize the time** clock; if the computer system does not have an active connection to the network, then: determine whether a default network connection mode of the computer system requires a user-disruptive action; if the default network connection mode requires a user-disruptive action, then wait until the **computer** system has an active connection to the network to access the network to **synchronize the time** clock; and if the default connection mode does not require a user-disruptive action, then: immediately establish an active connection to the network; and access the network to synchronize the time clock.

Claims Text - CLTX (31):

31. A **machine**-readable storage medium tangibly embodying sequences of instructions executable by a processing system to perform operations for **synchronizing a time** clock maintained by the processing system, the operations comprising: determining whether a network connection mode of the processing system is user disruptive; accessing the network to synchronize the time clock only when the processing system has an active connection to the network if the network connection mode is user-disruptive; and immediately establishing an active connection to the network to synchronize the time clock if the connection mode is not user-disruptive and the processing system does not have an active connection to the network.

Other Reference Publication - OREF (1):

Savage et al, Real-Time Mach **Timers**: Exporting Time to the User, www.cs.washington.edu/homes/savage/papers/machnix93.ps.*